Class Notes 19:

Numerical Methods (2/2)

82 – Engineering Mathematics

Runge-Kutta Methods

- The slope function of f is replaced by a weighted average of slopes over the interval $x_n \le x \le x_{n+1}$

$$y_{n+1} = y_n + h(\underbrace{w_1 k_1 + w_2 k_2 + \cdots + w_m k_m}_{\text{weighted average}})$$

$w_i - (i = 1, 2, \cdots, m)$ – constants generally satisfy

$$w_1 + w_2 + \cdots + w_m = 1$$

$k_i - (i = 1, 2, \cdots, m)$ – the function f evaluated at selected point (x, y) for which $x_n \le x \le x_{n+1}$

m – The order of the method

Runge-Kutta Methods

For
$$\left.\begin{aligned} m &= 1 \\ w_1 &= 1 \\ k_1 &= f(x_n, y_n) \end{aligned}\right\} \rightarrow \begin{array}{c} y_{n+1} = y_n + hf(x_n, y_n) \\ First\ order\ RK\ (RK1) \end{array}$$

Second order RK method

$$y_{n+1} = y_n + h(w_1 k_1 + w_2 k_2)$$

$$\begin{cases} k_1 = f(x_n, y_n) \\ k_2 = f(x_n + \alpha h, y_n + \beta h k_1) \end{cases}$$

$$\begin{cases} w_1 + w_2 = 1 & & w_1 = 1 - w_2 \\ w_2 \alpha = 1/2 & \rightarrow & \alpha = 1/2w_2 \\ w_2 \beta = 1/2 & & \beta = 1/2w_2 \end{cases}$$

Taylor polynomial degree 2

$$y(x_{n+1}) = y(x_n + h) =$$

$$= y(x_n) + hy'(x_n) + \frac{h^2}{2!} y''(x_n)$$

Runge-Kutta Methods

The choice $w_2 = \dfrac{1}{2}$ (can be chosen arbitrarily) $\rightarrow \begin{cases} w_1 = 1/2 \\ \alpha = 1 \\ \beta = 1 \end{cases}$

$$y_{n+1} = y_n + \frac{h}{2}(k_1 + k_2)$$

$$\begin{cases} k_1 = f(x_n, y_n) \\ k_2 = f(x_n + h, y_n + hk_1) \end{cases}$$

since

$$x_n + h = x_{n+1}$$

$$y_n + hk_1 = y_n + hf(x_n, y_n)$$

$\rightarrow$ Improved Euler's method

Fourth-order Runge-Kutta Methods (RK4)

$$y_{n+1} = y_n + h(w_1 k_1 + w_2 k_2 + w_3 k_3 + w_4 k_4)$$

$$\begin{cases} k_1 = f(x_n,\ y_n) \\ k_2 = f(x_n + \alpha_1 h,\ y_n + \beta_1 h k_1) \\ k_3 = f(x_n + \alpha_2 h,\ y_n + \beta_2 h k_1 + \beta_3 h k_2) \\ k_4 = f(x_n + \alpha_3 h,\ y_n + \beta_4 h k_1 + \beta_5 h k_2 + \beta_6 h k_3) \end{cases}$$

$$\begin{cases} 11 \text{ Equations} \\ 13 \text{ Unkown} \end{cases}$$

Taylor polynomial degree 4

$$y(x_{n+1}) = y(x_n + h) = y(x_n) + hy'(x_n) + \frac{h^2}{2!} y''(x_n) + \frac{h^3}{4!} y'''(x_n) + \frac{h^4}{5!} y^{(4)}(x_n)$$

Fourth-order Runge-Kutta Methods (RK4)

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$\begin{cases} k_1 = f(x_n,\ y_n) \\ k_2 = f\left(x_n + \frac{1}{2}h,\ y_n + \frac{1}{2}hk_1\right) \\ k_3 = f\left(x_n + \frac{1}{2}h,\ y_n + \frac{1}{2}hk_2\right) \\ k_4 = f\left(x_n + h,\ y_n + hk_3\right) \end{cases}$$

Fourth-order Runge-Kutta Methods (RK4)

For $y' = f(t, y), \;\; y(t_0) = y_0$

RK4 algorithm

$$
\begin{aligned}
k1 &= f(t, y) \\
k2 &= f(t + 0.5 \cdot h, \; y + 0.5 \cdot h \cdot k1) \\
k3 &= f(t + 0.5 \cdot h, \; y + 0.5 \cdot h \cdot k2) \\
k4 &= f(t + h, \; y + h \cdot k3) \\
y &= y + (h/6) \cdot (k1 + 2 \cdot k2 + 2 \cdot k3 + k4) \\
t &= t + h
\end{aligned}
$$

- RK4 method is relatively simple to use and sufficiently accurate to handle many problems efficiently.

Runge-Kutta Methods - Error

Method	Local truncation	Global truncation
Euler	h^2	h
Improved Euler	h^3	h^2
Runge-Kutta Second order	h^3	h^2
Runge-Kutta Fourth order	h^5	h^4
Adams-Bashforth-Moulton	h^5	h^4

Runge-Kutta Methods - Example

$$y' = 1 - t + 4y, \; y(0) = 1$$

Taking $h = 0.2$ for $RK4$ method,

$$k_{01} = f(0, 1) = 5$$

$$k_{02} = f(0 + 0.1, 1 + 0.5) = 6.9$$

$$k_{03} = f(0 + 0.1, 1 + 0.69) = 7.66$$

$$k_{04} = f(0 + 0.2, 1 + 1.532) = 10.928$$

Thus,

$$y_1 = 1 + \frac{0.2}{6}[5 + 2(6.9) + 2(7.66) + 10.928]$$

$$= 2.5016$$

Runge-Kutta Methods - Example

$$y' = 1 - t + 4y,\ y(0) = 1$$

Compare results

	Improved Euler	Runge-Kutta			Exact
t	h=0.025	h=0.2	h=0.1	h=0.05	
2.0	3496.6702	3490.5574	3535.8667	3539.8804	3540.2001
error	1.23%	1.40%	0.122%	0.00903%	
evaluation #	160	40			

Shortcoming with Fixed Step Size

- A step size that is small enough in some parts of the interval can be not enough to others.

- Local truncation error can be estimated in each step and change step size accordingly.

  - ~~One~~ way to estimate error is calculating difference between fourth order and fifth order method results

Multistep Methods

- In previous methods, the only data at $t = t_n$ are used to calculate an approximate value at $t = t_{n+1}$. (one-step methods)

$$\phi_n \rightarrow \phi_{n+1}$$

- What if we use a few points rather than just the value at the last point? (Multistep Methods)

$$\cdots, \phi_{n-1}, \phi_n \rightarrow \phi_{n+1}$$

Multistep Methods

Adams-Bashforth method

$$\phi(t_{n+1}) - \phi(t_n) = \int_{t_n}^{t_{n+1}} \underline{\phi'(t)} dt$$

↑

Approximate this term by a polynomial of degree k
(Higher degree gives more accuracy)

Example, with a first degree polynomial(k=1)

$$P_1(t) = At + B \longrightarrow P_1(t_n) = f(t_{n,} y_n) \longrightarrow At_n + B = f_n$$

$$P_1(t_{n-1}) = f(t_{n-1,} y_{n-1}) \qquad At_{n-1} + B = f_{n-1}$$

$$\longrightarrow A = \frac{f_n - f_{n-1}}{h} \longrightarrow \text{Integration} \longrightarrow y_{n+1} = y_n + \frac{3}{2} h f_n - \frac{1}{2} h f_{n-1}$$

$$B = \frac{f_{n-1} t_n - f_n t_{n-1}}{h}$$

Multistep Methods

Adams-Moulton method

- Principle is similar with Adams-Bashforth method, but use different points. Use (t_n, y_n) and (t_{n+1}, y_{n+1})

Result for k =1 case

$$\underline{y_{n+1}} = y_n + \frac{1}{2} h f_n + \frac{1}{2} h f(t_{n+1}, \underline{y_{n+1}})$$

implicit

- Adams-Moulton formulas of moderate order are considerably more accurate than Adams-Bashforth method. But, implicit.

Multistep Methods

Adams-Bashforth-Moulton method

- Combining two formulas to achieve both simplicity and accuracy.

(Predictor/corrector method)

For k=4,

Predictor(Adams-Bashforth)

$$y^*_{n+1} = y_n + \frac{h}{24}(55y'_n - 59y'_{n-1} + 37y'_{n-2} - 9y'_{n-3})$$

Corrector(Adams-Moulton)

$$y'_{n+1} = f(x_{n+1}, y_{n+1})$$

$$y_{n+1} = y_n + \frac{h}{24}(9f_{n+1} + 19f_n - 5f_{n-1} + f_{n-2})$$

Multistep Methods

- In order to use multistep methods, it is necessary first to calculate a few y by some other method.

- One way to proceed is to use a one-step method of comparable accuracy to calculate the necessary starting values.

Multistep Methods - Example

$$y' = x + y - 1$$

$$y(0) = 1$$

$$h = 0.2$$

$y(0.8)$ – will be approximated by y_4

Use RK4 method with $x_0 = 0;\ y_0 = 1;\ h = 0.2$ to obtain

$$y_1 = 1.02140000$$

$$y_2 = 1.09181796$$

$$y_3 = 1.22210646$$

For $x_0 = 0,\ x_1 = 0.2;\ x_2 = 0.4;\ x_3 = 0.6$

$$y'_0 = f(x_0, y_0) = 0 + 1 - 1 = 0$$

$$y'_1 = f(x_1, y_1) = 0.2 + 1.0210000 - 1 = 0.22140000$$

$$y'_2 = f(x_2, y_2) = 0.4 + 1.09181796 - 1 = 0.49181796$$

$$y'_3 = f(x_3, y_3) = 0.6 + 1.22210646 - 1 = 0.82210646$$

Multistep Methods - Example

$$y_4^* = y_3 + \frac{0.2}{24}(55y_3' - 59y_2' + 37y_1' - 9y_0') = 1.42535975$$

Use the corrector

$$y' = f(x_4, y_4^*) = 8.8 + 1.42535975 - 1 = 1.22535975$$

$$y_4 = y_3 + \frac{0.2}{24}(9y_4' + 19y_3' - 5y_2' + y_1') = 1.42552788$$

The actual value of $y(0.8) = 1.42554093$

Stability of Numerical Methods

Stable – Small changes in the initial condition result in only small changes in the computed solution

Unstable – If it is not stable

-In each step after the first step of a numerical technique, we are starting over again with a new initial value problem, when the initial condition is the approximate solution value computed in the previous step.

-Because of the presence of round-off error, this value will almost certainly vary at least slightly from the true value of the solution.

Stability of Numerical Methods

Detecting instability : 1. Decrease the step size
2. Perturb the initial conditions

$$y(0) = 1 \rightarrow y(0) = 0.9999$$

Higher-order Equation and System

- Second order initial value problem

$$y'' = f(x, y, y')$$

$$I.C. \begin{cases} y(x_0) = y_0 \\ y'(x_0) = u_0(x_0) \end{cases}$$

- Convert the problem into a set of first order diff. eq.

$$y' = u$$

$$u' = f(x, y, u)$$

since $y'(x_0) = u(x_0) \rightarrow I.C. \begin{cases} y(x_0) = y_0 \\ u(x_0) = u_0 \end{cases}$

Higher-order Equation and System

- Apply a particular numerical method to each first order diff. eq. in the system

For example - Euler's method

$$y_{n+1} = y_n + hu_n$$
$$u_{n+1} = u_n + hf(x_n, y_n, u_n)$$

For example - RK4

$$y_{n+1} = y_n + \frac{h}{6}(m_1 + 2m_2 + 2m_3 + m_4)$$
$$u_{n+1} = u_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$m_1 = u_n$	$k_1 = f(x_n, y_n, u_n)$
$m_2 = u_n + 1/2 \cdot hk_1$	$k_2 = f(x_n + 1/2 \cdot h,\ y_n + 1/2 \cdot hm_1,\ u_n + 1/2 \cdot hk_1)$
$m_3 = u_n + 1/2 \cdot hk_2$	$k_3 = f(x_n + 1/2 \cdot h,\ y_n + 1/2 \cdot hm_2,\ u_n + 1/2 \cdot hk_2)$
$m_4 = u_n + 1/2 \cdot hk_3$	$k_4 = f(x_n + h,\ y_n + hm_3,\ u_n + hk_3)$

Higher-order Equation and System - Example

Euler's method

$$y'' + xy' + y = 0 \;\rightarrow\; y'' = -xy' - y$$

$$IC \begin{cases} y(0) = 1 \\ y'(0) = 2 \end{cases}$$

$$y' = u$$

$$u' = -xu - y$$

$$\begin{cases} y(0) = 1 \\ u(0) = 2 \end{cases}$$

Use step size $h = 0.1;\; y_0 = 1;\; u_0 = 2$

$$\begin{cases} y_1 = y_0 + hu_0 = 1 + (0.1)2 = 1.2 \\ u_1 = u_0 + (h)[-x_0 u_0 - y_0] = 2 + (0.1)[-0.2 - 1] = 1.9 \end{cases}$$

$$\begin{cases} y_2 = y_1 + hu_1 = 1.2 + (0.1)1.9 = 1.39 \\ u_2 = u_1 + (h)[-x_1 u_1 - y_1] = 1.9 + 0.1[-(0.1)(1.9) - 1.2] = 1.761 \end{cases}$$

$$y(0.1) = 1.2$$

$$y'(0.1) = 1.9$$

$$y(0.2) \cong 1.39$$

$$y'(0.2) \cong 1.761$$

System Reduction to First-order System

High order system of diff eq. $\xrightarrow{\text{(Reduce)}}$ First order system of dff. Eq.

Example

$$\begin{cases} x'' - x' + 5x + 2y'' = e^t \\ -2x + y'' + 2y = 3t^2 \end{cases}$$

$$\begin{cases} x'' + 2y'' = e^t - 5x - x' \\ y'' = 3t^2 + 2x - 2y \end{cases}$$

System Reduction to First-order System

$$x'' = -9x + 4y + x' + e^t - 6t^2$$

Since the second equation of the system already express the highest order derivation of y

$$\begin{cases} x' = u \\ y' = v \\ u' = x'' = -9x + 4y + u + e^t - 6t^2 \\ v' = y'' = 2x - 2y + 3t^2 \end{cases}$$

Numerical Solution of a System

$$\begin{cases} \dfrac{dx_1}{dt} = g_1(t, x_1, x_2, \cdots, x_n) \\ \dfrac{dx_2}{dt} = g_2(t, x_1, x_2, \cdots, x_n) \\ \qquad \vdots \\ \dfrac{dx_n}{dt} = g_n(t, x_1, x_2, \cdots, x_n) \end{cases}$$

RK4

$$x' = f(t, x, y)$$

$$y' = g(t, x, y)$$

$$I.C. \begin{cases} x(t_0) = x_0 \\ y(t_0) = y_0 \end{cases}$$

Numerical Solution of a System

$$\begin{cases} x_{n+1} = x_n + \frac{h}{6}(m_1 + 2m_2 + 2m_3 + m_4) \\ y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \end{cases}$$

$$m_1 = f(t_n, x_n, y_n)$$
$$m_2 = f(t_n + 1/2 \cdot h, x_n + 1/2 \cdot hm_1, y_n + 1/2 \cdot hk_1)$$
$$m_3 = f(t_n + 1/2 \cdot h, x_n + 1/2 \cdot hm_2, y_n + 1/2 \cdot hk_2)$$
$$m_4 = f(t_n + h, x_n + hm_3, y_n + hk_3)$$

$$k_1 = g(t, x_n, y_n)$$
$$k_2 = g(t_n + 1/2 \cdot h,\ x_n + 1/2 \cdot hm_1,\ y_n + 1/2 \cdot hk_1)$$
$$k_3 = g(t_n + 1/2 \cdot h,\ x_n + 1/2 \cdot hm_2,\ y_n + 1/2 \cdot hk_2)$$
$$k_4 = g(t_n + h,\ x_n + hm_3,\ y_n + hk_3)$$

Numerical Solution of a System - Example

RK4 method

$$x' = 2x + 4y \qquad\qquad I.C. \begin{cases} x(0) = -1 \\ y(0) = 6 \end{cases}$$
$$y' = -x + 6y$$

Find x(0.6), y(0.6) and compare values for h = 0.2, 0.1

For $h = 0.2$

$$f(t, x, y) = 2x + 4y$$
$$g(t, x, y) = -x + 6y$$

$$m_1 = f(t_0, x_0, y_0) = f(0, -1, 6) = 2(-1) + 4(6) = 22$$
$$k_1 = g(t_0, x_0, y_0) = g(0, -1, 6) = -1(-1) + 6(6) = 37$$
$$m_2 = f(t_0 + 1/2 \cdot h, x_0 + 1/2 \cdot hm_1, y_0 + 1/2 \cdot hk_1) = f(0.1,\ 1.2,\ 9.7) = 41.2$$
$$k_2 = g(t_0 + 1/2 \cdot h, x_0 + 1/2 \cdot hm_1, y_0 + 1/2 \cdot hk_1) = g(0.1,\ 1.2,\ 9.7) = 57$$
$$m_3 = f(t_0 + 1/2 \cdot h, x_0 + 1/2 \cdot hm_2, y_0 + 1/2 \cdot hk_2) = f(0.1,\ 3.12,\ 11.7) = 53.04$$
$$k_3 = g(t_0 + 1/2 \cdot h, x_0 + 1/2 \cdot hm_2, y_0 + 1/2 \cdot hk_2) = g(0.1,\ 3.12,\ 11.7) = 67.08$$
$$m_4 = f(t_0 + h, x_0 + hm_3, y_0 + hk_3) = f(0.2,\ 9.608,\ 19.416) = 96.88$$
$$k_4 = g(t_0 + h, x_0 + hm_3, y_0 + hk_3) = g(0.2,\ 9.608,\ 19.416) = 106.88$$

Numerical Solution of a System - Example

$$x_1 = x_0 + \frac{0.2}{6}(m_1 + 2m_2 + 2m_3 + m_4)$$

$$= -1 + \frac{0.2}{6}(22 + 2(41.2) + 2(53.04) + 96.88) = 9.2453$$

$$y_1 = y_0 + \frac{0.2}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$= 6 + \frac{0.2}{6}(37 + 2(57) + 2(67.08) + 106.888) = 19.0683$$

$t = 0.6$

h	x(0.6)	y(0.6)
0.1	160.7563	152.0025
0.2	158.9430	150.8192
Exact	160.9384	152.1198

Exact →

$$x(t) = (26t - 1)e^{4t}$$

$$y(t) = (13t + 6)e^{4t}$$